

HP OpenView

Storage Mirroring application notes

High availability for Microsoft SQL Server 7.0

Legal and notice information

© Copyright 2005-2006 Hewlett-Packard Development Company, L.P.

Hewlett-Packard Company makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

This document contains proprietary information, which is protected by copyright. No part of this document may be photocopied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard. The information is provided "as is" without warranty of any kind and is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft, Windows, Windows NT, and Windows XP are U.S. registered trademarks of Microsoft Corporation.

Storage Mirroring High availability for Microsoft SQL Server 7.0 application notes

Document overview

This document is a Storage Mirroring application note. An application note provides guidelines on the use of Storage Mirroring in a specific environment.

This document contains:

- **Document overview**—Explains what an application note contains, how it should be used, what you need to know before trying to use the application note, and where you can go for more information.
- **Solution overview**—Explains how the application works with Storage Mirroring and describes the considerations that you must weigh when implementing your Storage Mirroring solution. Review this section to make sure that you understand the theory involved with using Storage Mirroring and your application. Includes both basics, such as system requirements, as well as configuration and environment-specific topics, such as interactions with specific clients or special considerations for WAN (Wide Area Network) environments. Pay special attention to those topics that are directly related to your environment.
- **Sample implementations**—Describes specific examples of how to use Storage Mirroring for this solution. This includes information about the specific system setup used in the sample implementation. Use these procedures as a guideline for creating your own implementation. Because no two environments or configurations are exactly the same, you will probably need to implement additional or different steps than what is documented here in order to make the solution work in your environment.

Audience

This document is written for network and application administrators who have a working understanding of the applications and environments where the Storage Mirroring solution is to be deployed. You may need to expand on the documented information in order to customize the solution to fit your environment.

Before you use this application note, you should have an understanding of:

- Storage Mirroring
- SQL Server

Expectations

Application notes are intended to provide a framework for configuring a Storage Mirroring solution in a specific environment and to draw attention to decisions you will need to make when configuring your solution.

Because there are an infinite number of possible configuration, network, and environment scenarios, application notes contain general configuration guidelines as well as an example configuration procedure that has been tested for a specific environment.

This document assumes that you are comfortable working with your operating system, Storage Mirroring, and SQL Server.

Related documentation

Before you begin to configure your solution, make sure that you have complete documentation for your operating system, application, and Storage Mirroring. This application note does not provide step-by-step instructions for using standard operating system, application, and Storage Mirroring functionality.

The following documents contain additional information that you may need while setting up this solution:

- *HP OpenView Storage Mirroring user guide* or online documentation
- Reference guides or documentation for SQL Server

Getting help


Hewlett-Packard has application notes that describe how to configure Storage Mirroring with a variety of popular third-party applications. These application notes are available on the Storage Mirroring web site: <http://h18006.www1.hp.com/products/storage/software/sm/index.html>.

Solution overview

Microsoft SQL Server is a scalable, reliable, flexible, and high-performance relational database management system for Microsoft Windows 200x server-based systems. Storage Mirroring provides real-time enterprise data protection and replication. Storage Mirroring can be used to provide high availability for your SQL Server.

This document describes the steps necessary to configure Storage Mirroring to provide high availability for Windows 200x servers running Microsoft SQL Server 2000. These procedures allow a secondary server to assume the identity and role of a failed SQL Server while maintaining the availability of SQL services with minimal disruption or data loss.

To complete these instructions, you will install SQL Server 2000 and Storage Mirroring, and configure Storage Mirroring for replication and failover. Due to the complexities of these applications, this document is intended for network administrators with experience installing, configuring, and maintaining network applications including Storage Mirroring and Microsoft SQL Server.

 **NOTE:** Storage Mirroring allows you to configure one target to monitor and failover for one or more source machines. In a one-to-one configuration, you will want to replicate your SQL data to the same location on the target so that failover is automatic. In a many-to-one configuration, each SQL data store will need to be replicated to a unique location and then renamed to the corresponding SQL directory on the source before failover occurs.

Modifying the sample script files

After you modify the sample scripts, save them with a new name to remove the `SAMPLE_` prefix. Copy the scripts to the directory where Storage Mirroring is installed.

The sample batch files provided are only examples. Because no two environments or configurations are exactly the same, you **MUST** modify the sample scripts in order to make the solution work in your environment.

Configuring the replication set

For a default SQL installation, you *must* include the SQL Data directory in your replication set (`<drive>:\Program Files\Microsoft SQL Server\MSSQL\Data`). This directory includes the SQL application data and transaction log files.

In addition, you *may* need to include the following directories, depending on your environment configuration:

- `Tempdb` files. You should always include the `tempdb` files, unless you can determine that they are not being used by any application. Some applications write data to the `tempdb` file.
- Any other directories (even if on different drives) that you may have created to store SQL data files.
- The SQL Log directory (`<drive>:\Program Files\Microsoft SQL Server\MSSQL\Log`). This directory includes the SQL error logs from the source, which may be useful when determining the cause of a source failure. In order to prevent overwriting the target's error logs, you will need to replicate the log files to a different location on the target. Create a separate replication set for the error logs and select the "All to One" mapping option.

You *do not* need to replicate the application files, since they already exist on the target machine.

Enabling compression

By enabling compression, you can reduce the amount of bandwidth needed to transmit Storage Mirroring data. When compression is enabled, the data is compressed before it is transmitted from the source. When the target receives the compressed data, it uncompresses it and then writes it to disk. On a default Storage Mirroring connection, compression is disabled.

Because the files that should be included in a SQL replication set can generate a significant amount of data, you should enable compression for the connection. For more information about enabling compression, see the *HP OpenView Storage Mirroring user's guide*. However, keep in mind that the process of compressing data impacts processor usage. If you notice an impact on performance while compression is enabled in your environment, either adjust to a lower level of compression, or leave compression disabled.

Configuring SQL memory usage

Storage Mirroring uses memory to queue operations and data on both the source and target. Since the source server is typically running a production application, it is important that the amount of memory Storage Mirroring and the other applications use does not exceed the amount of RAM in the system. If the applications require more memory than there is RAM, the system will begin to swap pages of memory to disk and the system performance will degrade.

For instance, SQL Server will use all of the available system memory when needed by default, and it may use almost all of the system memory during high-load operations. These high-load operations are precisely what cause Storage Mirroring to need memory to queue the data being changed by SQL Server. On a server with 1 GB of RAM running SQL Server and Storage Mirroring, you might configure SQL Server to use only 512 MB and Storage Mirroring to use 256 MB, leaving 256 MB for the operating system and other applications on the system. Many other server applications will use almost all system memory by default, so it is important to check and configure applications appropriately, particularly on high-capacity servers.

Sample Implementation

This section describes an example of how to configure Storage Mirroring and SQL Server. Use these procedures as a guideline for creating your own implementation. Because no two environments or configurations are exactly the same, you will probably need to implement additional or different steps than what is documented here in order to make the solution work in your environment.

Requirements

- Two servers that meet one of the following operating system requirements:
 - Microsoft Windows NT 4.0 with Service Pack 4 or higher
 - Microsoft Windows 200x



NOTE: The two servers should both be running the same operating system. Although cross-platform mirroring and replication are available, HP recommends that the two servers be the same platform for effective failover and failback.

-
- It is recommended that both source and target servers be member servers. (You may experience problems with promotion and demotion during failover if either of the machines are Primary or Backup Domain Controllers.)
 - Both servers must be connected to the same physical network
 - Two licensed copies of Microsoft SQL Server 7.0

- Two licensed copies of Storage Mirroring

Install software on the source

1. Install Microsoft SQL Server on the source, if it is not already installed.
2. Record the drive and directory where Microsoft SQL is installed. By default, this is <drive>:\MSSQL7.

SQL Installation Drive and Directory: _____

3. Install Storage Mirroring on the source machine using the installation defaults. See the *HP OpenView Storage Mirroring getting started guide* for details.

Install and configure software on the target

1. Install Storage Mirroring on the target using the installation defaults. See the *HP OpenView Storage Mirroring getting started guide* for details.
2. In **Control Panel, Services**, double-click the Storage Mirroring service.
3. Mark the check box **Allow Service to Interact with Desktop** and click **OK**.
4. Install Microsoft SQL on the target using the same drive and directory specifications recorded in step 2 of the previous section.
5. Set the SQL services to manual startup so that all SQL files are closed on the target and the Storage Mirroring source can replicate the changes. The service names are dependent on the operating system you are using.



NOTE: If a failure should occur, the failover and failback scripts that you will be creating will control the stopping and starting of the SQL services.

- **Windows 200x**—Distributed Transaction Coordinator, Message Queuing, MSSQLServer, and SQLServerAgent
- **Windows NT 4.0**—MSDTC, Message Queuing, MSSQLServer, and SQLServerAgent

Configure and begin mirroring and replication

1. Select **Start, Programs, Storage Mirroring, Management Console**.
2. Double-click your source machine to log on.
3. Right-click the source and select **Properties**.
4. On the Source tab, enable **Block Checksum All Files on a Difference Mirror** and click **OK**.
5. Right-click your source machine and select **New, Replication Set** and enter the desired name for the replication set.
6. Select the SQL data you wish to protect. For more information about what you should include in your SQL replication set, see "[Configuring the replication set](#)" on page 4.
7. Right-click the replication set name and select **Save** to save the replication set.
8. Drag and drop the replication set onto the target. The Connection Manager dialog box opens.
9. The **Source Server**, **Target Server**, **Replication Set**, and **Route** fields will automatically be populated. If you have multiple IP addresses on your target, verify the **Route** field is set to the correct network path. (For detailed information on connecting a source and target, see the *HP OpenView Storage Mirroring user's guide*.)

10. Select **One to One** to map the replication set data from the source to an identical volume/directory structure on the target.

11. On the Orphans tab, select the **Move/Delete Orphan Files** checkbox.

12. On the Mirroring tab, select the type of mirror, either **Full** or **File Differences**, to perform.

NOTE: If the target has been previously mirrored to or restored, select File differences with the Use block checksum option so that only the changed data is sent across the network.

13. Click **Connect** to start the mirror and replication processes.

NOTE: If you start SQL Server and mount the replicated databases on the target, or if the data on the target is otherwise modified, the data on the source and target will no longer match. If the updated data on the target is not needed, perform a full or difference with block checksum mirror from the source to the target. If the updated data on the target is needed, restore the data from the target to the source.

Configure failover and begin failure monitoring

The following steps should be performed on the target machine.

1. If a failure occurs, you will want to have the SQL services start on the target machine automatically. To do this, create a batch file called `postover.bat` using the sample batch file below. Save the batch file to the same directory where your Storage Mirroring files are installed.

NOTE: After you modify the sample scripts, save them with a new name to remove the `sample_` prefix. Copy the scripts to the directory where Storage Mirroring is installed.

The sample batch files provided are only examples. Because no two environments or configurations are exactly the same, you **MUST** modify the sample scripts in order to make the solution work in your environment.

SAMPLE POSTOVER.BAT

```
rem This file is configured by default to run with SQL 7.0 on Windows 200x
rem You may need to comment the default lines out and uncomment the set
rem lines for the Windows version you are using

rem If you are using SQL 7.0 and Windows 200x use the following four commands.
net start "Distributed Transaction Coordinator"
net start "Message Queuing"
net start "MSSQLServer"
net start "SQLServerAgent"

rem If you are using SQL 7.0 and Windows NT 4.0 use the following four commands.
rem net start "MSDTC"
rem net start "Message Queuing"
rem net start "MSSQLServer"
rem net start "SQLServerAgent"
```

2. After a failure is resolved, you will be ready to bring your source back online. At this time, you will want to stop the SQL services on the target automatically. To do this, create a batch file called `preback.bat` using the sample batch file below. Save the batch file to the same directory where your Storage Mirroring files are installed.

SAMPLE PREBACK.BAT

```
rem This file is configured by default to run with SQL 7.0 on Windows 200x
rem You may need to comment the default lines out and uncomment the set of lines for the Windows version
rem you are using

rem If you are using SQL 7.0 and Windows 200x use the following four commands.
net stop "Distributed Transaction Coordinator"
net stop "Message Queuing"
net stop "SQLServerAgent"
net stop "MSSQLServer" /y

rem If you are using SQL 7.0 and Windows NT 4.0 use the following four commands.
rem net stop "MSDTC"
rem net stop "Message Queuing"
rem net stop "SQLServerAgent"
rem net stop "MSSQLServer" /y
```

3. Select **Start, Programs, Storage Mirroring, Failover Control Center**.
4. Select the target machine from the list of available machines. If the target you need is not displayed, click **Add Target**, enter the machine name, and click **OK**.
5. To add a monitor for the selected target, click **Add Monitor**. Type the name of the source machine and click **OK**. The Monitor Settings window will open.
6. In the Monitor Settings window, mark the IP address that is going to failover.
7. Click Scripts and specify the scripts that were created earlier, using postover.bat for the target post-failover script and preback.bat for the target pre-failback script.

NOTE: The scripts are processed using the same account running the Storage Mirroring service.

8. Click **OK** to go back to the Monitor Settings dialog box.
9. Click **OK** to begin monitoring the source machine.

In the event of a source machine failure, your target machine is now ready to stand in for your source. For information on monitoring failover, see the *HP OpenView Storage Mirroring user's guide*.

Restoring your SQL data

If your source experiences a failure, such as a power, network, or disk failure, your target machine will stand in for the source while you resolve the source machine issues. During the source machine downtime, data is updated on the target machine. When your source machine is ready to come back online, the data is no longer current and must be updated with the new data on the target machine.

1. Verify that your source machine is not connected to the network. If it is, disconnect it.
2. Resolve the source machine problem that caused the failure.



NOTE: If you must rebuild your hard drive, continue with step 3. If you do not need to rebuild your hard drive, continue with step 8.

3. Install Windows. Since your source machine is not connected to the network, go ahead and use the source's original name and IP address.
4. Install Storage Mirroring using the installation defaults.
5. Install SQL using the same drive and directory settings recorded in step 2 of the first section.
6. Stop SQL services on the target and set them to manual startup so that all SQL files are closed on the target and the Storage Mirroring source can replicate the changes. The service names are dependent on the operating system you are using.

- **Windows 200x**—Distributed Transaction Coordinator, Message Queuing, MSSQLServer, and SQLServerAgent
 - **Windows NT 4.0**—MSDTC, Message Queuing, MSSQLServer, and SQLServerAgent
7. Rename any log files located in SQL log directory. By default, this is <drive>:\MSSQL7\Log.
 8. **Verify that SQL is not running on the source.** The SQL services must not be running at this time. Depending on the type of failure, your services may be set to manual startup but could still be running. **Stop your SQL services and set them to manual.**
 9. Select **Start, Programs, Storage Mirroring, Failover Control Center.**
 10. Select the target machine that is currently standing in for the failed source.
 11. Select the failed source and click **Failback.**

The pre-failback script entered during the failover configuration stops the SQL services on the target so that no additional changes can be made.
 12. You will be prompted to determine if you want to continue monitoring the source server. Do not choose **Continue** or **Stop** at this time.
 13. Connect the source machine to the network.
 14. After the source is back online, select whether or not you want to continue monitoring this source machine (**Continue** or **Stop**).

NOTE: Verify that the Storage Mirroring connection on the source has been disconnected (right-click the connection in the Storage Mirroring Management Console and select **Disconnect**).

15. To begin the restoration process, open the Storage Mirroring Management Console and select **Tools, Restoration Manager.**

NOTE: You can also run the Storage Mirroring DTCL automated restoration script, which can be found in the *HP OpenView Storage Mirroring user's guide*, to complete the remaining steps in this section.

16. Complete the appropriate fields as described below.
 - **Original Source**—The name of the source machine where the data originally resided.
 - **Restore From**—The name of the target machine that contains the replicated data.
 - **Replication Set**—The name of the replication set to be restored.
 - **Restore To**—The name of the machine where the data will be restored. This may or may not be the same as the original source machine.
17. Identify the correct drive mappings for the data and any other restoration options necessary. For detailed information on the restoration options, see the *Storage Mirroring User's Guide*.
18. Verify that the selections you have made are correct and click **Restore**. The restoration procedure time will vary depending on the amount of data that you have to restore.
19. After the restoration is complete, start the SQL services on the source machine.
20. Reestablish the Storage Mirroring SQL replication set connection.
21. On the Mirroring tab, select **File differences** with the Use block checksum option.

NOTE: If the target has been previously mirrored to or restored, select **File differences** with the Use block checksum option so that only the changed data is sent across the network.

At this time, your data is restored back to your source machine, the source machine is again the primary SQL server, and, if you selected to continue failover monitoring, the target is available to stand in for the source in the event of a failure.